Avoiding DAD for Improving Real-Time Communication in MIPv6 Environments

Marcelo Bagnulo, Ignacio Soto, Alberto García-Martínez, Arturo Azcorra.

Universidad Carlos III de Madrid – Dep. Ingeniería Telemática
Avenida Universidad, 30. Leganés, Madrid 28911 – España
{marcelo,isoto,alberto,azcorra}@it.uc3m.es

Abstract. Current specification of address configuration mandates the execution of the Duplicate Address Detection (DAD) mechanism to prevent address duplication. However, a proper support for real time multimedia applications in mobile IPv6 nodes is undermined by the disruption imposed by DAD. In order to overcome this limitation, the usage of randomly generated IPv6 Interface Identifiers without previously performing DAD is proposed, based on the statistic uniqueness of the addresses generated through this method. The address duplication risk is quantified through the calculation of the probability of an Interface Identifier collision among the nodes sharing a link. The calculated probability is deemed to be negligible compared to other causes of communication failure, such as network outages.

1 Introduction[1]

In order to take full advantage of multimedia capabilities of current mobile devices, the network infrastructure must provide an uninterrupted flow of information to appropriately support real time traffic. However, the requirement for performing Duplicate Address Detection (DAD) in the address autoconfiguration mechanism limits the performance of mobility in IPv6, provided by Mobile IPv6 [1]. Using this protocol, a Mobile Node (MN) that joins a subnet must configure an on-link address in that subnet, the Care-of-Address (CoA), before being able to communicate. According to the Stateless Address Autoconfiguration mechanism presented in [2], before using the CoA the MN must perform DAD for that address in order to guarantee its uniqueness on the link. It should be noted that DAD is a time consuming process. Although this it is not an issue for a desktop computer that is booting, the time required for DAD is critical in a mobile environment, since during this time the MN can not communicate and besides, all active communications of the MN are interrupted. The time required to perform DAD has a default value of one second [2], [3], a value subjectively deemed as not acceptable for interactive voice communications [4]. The Mobile IPv6 (MIPv6) specification [1] identifies the aforementioned problem and states that a MN can decide not to perform DAD,

[1] This research was supported by the LONG (Laboratories Over Next Generation networks) project, IST-1999-20393 and MobyDick (Mobility and Differentiated Services in a Future IP Network) project, IST-2000-25394 .

pointing this as a trade-off between safety and the time needed for the DAD procedure.

This document proposes the use of random numbers to create the Interface Identifier part of the IPv6 addresses, and assesses the risk of using these addresses without previously performing DAD. It should be noted that this solution is not restricted to a particular data-link layer technology, although it can be optimized in particular cases, such as GPRS, in which collision can be avoided by the GGSN (Gateway GPRS Support Node).

The remainder of the paper is structured as follows: in section 2 the risk of not using the DAD mechanism is quantified in several relevant scenarios. In section 3, implementation issues are discussed, including random number generation and security aspects. In the next section, alternative proposals are considered and finally, section 5 is devoted to conclusions.

2 Avoiding DAD: Risk Assessment

In this section we will asses the risk of using randomly generated Interface Identifiers (IIDs) in IPv6 aggregatable addresses [5] without previously performing DAD. In order to do that, we will quantify the probability of a duplicate address event in several relevant scenarios and we will compare it with the probability of other critical events.

2.1 Duplicate Address Event Probability Calculation and Bounding

In the considered hypothesis, the Interface Identifier part of the IPv6 address is generated randomly, meaning that the node will use a 64 bit long random number as the IID. Actually, only 62 bits of the IID will be generated randomly, since the IID´s semantics defined in [6] imposes that the u bit must be set to “local” and the g bit must be set to “individual”.

Considering that n is the number of possible IIDs (i.e. $n = 2^{62}$) and k is the number of interfaces (i.e. mobile nodes) on the same link, we will now calculate the probability of collision of two or more randomly generated IIDs:
We will represent the k IIDs in a link as a sequence of 62 bit long random variables I_i:

$I_1, I_2, ..., I_k$ sequence of random integer variables with uniform distribution between 1 and n ($k \ll n$)

We would like to obtain the probability that two or more I_is collide, i.e. $I_i = I_j$

The solution for this well known mathematical problem, called the "birthday problem", is presented in Appendix A.

The resulting expression for the probability of the collision of one or more I_i is:

$$P(n,k) = 1 - \frac{n!}{(n-k)!\,n^k} \quad (1)$$

In our particular case, $n = 2^{62}$, and k may vary depending on the considered scenario. We will now obtain an upper bound to $P(n,k)$ in order to simplify calculations (especially to avoid $n!$ computation)

Performing simple computations in equation (1), we easily obtain:

$$P(n,k) = 1 - \left(1 - \frac{1}{n}\right)\left(1 - \frac{2}{n}\right)\ldots\left(1 - \frac{k-1}{n}\right) \quad (2)$$

Since

$\forall i \in \{1,2,\ldots,k-1\} \quad \frac{i}{n} \le \frac{k-1}{n}$,

and considering that $k \le n$, then

$$1 - \left(1 - \frac{1}{n}\right)\left(1 - \frac{2}{n}\right)\ldots\left(1 - \frac{k-1}{n}\right) \le 1 - \left(1 - \frac{k-1}{n}\right)^{k-1}$$

Applying this last result to equation 2, we can obtain the following bound B:

$$P(n,k) \le \frac{n^{k-1} - (n-k+1)^{k-1}}{n^{k-1}} = B \quad (3)$$

We will next perform some calculations in order to quantify the order of magnitude of the probabilities involved:

We will bound $P(n,k)$ for the following values of *k*, which we consider to be representative of usual situations

$P(2^{62},20) \le 7.8e-17$

$P(2^{62},100) \le 2.1e-15$

$P(2^{62},500) \le 5.4e-14$

$P(2^{62},1000) \le 2.2e-13$

$P(2^{62},5000) \le 5.4e-12$

In order to fully seize the magnitude of the probabilities stated above, we can compare them with the probabilities of some rare events. For instance, according to Table 1.1 in [7], the probability of being killed by a lightning (per day) is about 1.1 e-10. Then, a mobile phone user should be more worried about being killed by a lightning in a given day than to have an interface identifier repeated when he performs a handoff.

Another relevant parameter that can be considered when evaluating the above probability, is the probability of a failure in a network device, since this failure would have similar effects i.e. the user can not continue the communication. So, the probability that a network device were not working properly in a given moment (when the user joins the network, for instance) can be calculated as follows:

$$P_{NEFails} ? \frac{MTTR}{MTBF ? MTTR}$$

Being MTTR the Meat Time To Repair and MTBF the Mean Time Between Failures. Good network equipment can have an MTBF of 300,000 hours and if we suppose that some backup device is available, the MTTR stands for the time needed to replace the faulty element, e.g. 0.1 hour (6 minutes). In this case, $P_{NEFails} ? 3.3e ? 7$.

We can see that $P_{NEFails}$ is several orders of magnitude higher than $P(n,k)$ in the cases calculated above.

2.2 Scenarios

We have quantified the probability of a collision of two or more IIDs. However, this probability is not the most relevant parameter when we try to evaluate and compare the probability of failure of the system, since a mobile user will join multiple links in a given period. So, it is relevant to quantify the probability of at least one collision when a user performs multiple handoffs.

As we stated above, $P(n,k)$ is the probability of a collision of two or more IIDs when there are k interfaces in the same link. Hence, it can be derived that the probability of having at least one collision after joining m links is:

$$P(n,k,m) ? 1 ? (1 ? P(n,k))^m \quad (4)$$

According to the bound B presented in equation 3 and considering that both $P(n,k)$ and B are lower than 1, we can infer the following bound:

$$P(n,k,m) ? 1 ? (1 ? B)^m \quad (5)$$

Therefore, in order to estimate the probability of a collision event during a given period, for instance a year, we must first establish a reasonable number of handoffs per day. If we consider 140 handoffs per day, which seems to be a considerable number, this would mean about 50.000 handoffs per year, i.e. m=50.000. Then the probability of having at least one collision over a year during which the mobile node has joined 140 links of 500 nodes per day is:

$$P(2^{62},500,50.000) ? 2.7e ? 9 \quad (6)$$

And, if the considered links have 5.000 nodes each, instead of 500, the probability is:

$$P(2^{62},5.000,50.000) ? 2.7e ? 7 \quad (7)$$

Considering that each time there is a collision there are two users affected, and not taking into account the collision of 3 or more IIDs for this estimation, there will be 6 users out of 1.000.000.000 that will have a communication problem during this year, if users make 140 handovers per day in networks containing 500 interfaces. In the case that users make 140 handovers per day in networks containing 5.000 interfaces, there will be 6 users out of 10.000.000 that will have a communication problem during this year. This probability could be contrasted with some network availability data provided by, for instance, mobile operators, but this data has proven to be extremely difficult to find.

3 Implementation Issues

In this section, we will address some implementation issues regarding random IIDs generation and related security concerns.

3.1 Random Numbers Generation

When considering the usage of random IIDs, the random number generation process must be properly addressed since it is essential to guarantee the statistic uniqueness of the identifier. Several methods have been proposed [8] to generate pseudo-random numbers based on information available in most platforms, such as laptops. However, in some cases, such as mobile phones, the resources required to perform appropriate random number generation may not be available. In such cases, it should be noted that it is not necessary to create the identifier in real time, as long as randomness were guaranteed. This means that when the node joins the network the identifier could have been created already in advance to a network change. It could even be pre-configured in the interface driver with the node using the same identifier without changing the probabilities calculations stated above; this is analogous to the day of birth in the birthday problem. This would reduce the complexity in the nodes, although a mechanism should be provided in order to solve recurrent collisions, caused for example, when two habitual users of the same segment collide.

3.2 Security Concerns

Randomly generated IIDs have also been considered in order to improve security. In particular, its usage has been proposed to ensure anonymity [9] and even to carry cryptographic information when Cryptographically Generated Addresses are used [10]. These proposals are fully compatible with the solution of this document so they can get the benefit of better performance by avoiding DAD.

4 Related Work

The requirement of performing DAD every time a node joins a link imposed by Stateless Address Autoconfiguration mechanisms [2] has already been stated to be a major limitation by the Internet community, resulting in several recommendations and solutions, some of which are discussed below.

The Mobile IP protocol [1] allows not performing DAD when a node joins a link in a trade-off between safety and the time needed for the DAD procedure. Besides, the use of fast handovers [11] allows performing DAD in advance, before the MN arrives to the subnet. In this case, the Access Router (AR) in the subnet is instructed to perform DAD on behalf of the MN before it enters the subnet. But then, the MN has to wait for the time needed to perform DAD before it can accomplish the handover. This is a problem when difficulties in the previous data-link layer connection force the handover. So we will again benefit from avoiding the DAD procedure.

Other solution, specific to 3GPP (3rd Generation Partnership Project), proposed to avoid DAD can be found in [12]. In this case DAD is not performed since every prefix is assigned to only one primary context, preventing from collisions by limiting the numbers of nodes that share the same link. However, this implies reserving a complete prefix of 64 bits to just a couple of interfaces, resulting in an enormous waste of address space.

5 Conclusions

The impact of the time required for DAD during handover on real time multimedia applications can not be underestimated. In this document we evaluate the usage of random numbers to construct the Interface Identifier part of IPv6 addresses and then we asses the risk of using such addresses without previously performing DAD. We conclude that the probability of failure due to a collision is low enough to be acceptable in most cases. Furthermore, we consider that the probability of a failure in a communication due to an IID collision is negligible compared with other quantifiable causes of failure, such as network equipment outage. It should be noted that in this estimation other relevant causes such as operation errors, which are usually considered to be more frequent than the ones presented, have not been included because of the unavailability of hard data. Other possible solutions for avoiding DAD have been presented along with its drawbacks, and we consider that the usage of random IIDs without previously performing DAD is an attractive option since it completely solves the presented problem and it does not incurs in excessive ulterior costs such as waste of address space.

References

1. Johnson, D. and C. Perkins, "Mobility Support in IPv6", Internet draft, Work in progress, July 2001.
2. Thomson, S. and T. Narten, "IPv6 Stateless Address Autoconfiguration", RFC 2462, December 1998.
3. Narten, T., Nordmark, E., Simpson, W., “Neighbor Discovery for IP Version 6 (IPv6)”, RFC 2461, December 1998
4. Gruber, J. and Strawczynski, L., “Subjective Effects of Variable Delay in Speech Clipping in Dynamically Managed Voice Systems,” IEEE Trans-actions on Communications, Vol. COM-33, No. 8, Aug. 1985.]
5. Hinden, R., O´Dell, M. and S. Deering, "An IPv6 Aggregatable Global Unicast Address Format", RFC 2374, July 1998.
6. Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 1998, 1998.
7. Schneier, B., "Applied cryptography", Wiley ISBN 0-471-12845-7,1996.
8. Eastlake, D., Crocker, S., Schiller, J., “Randomness Recommendations for security”, RFC 1750, December 1994
9. Narten, T., Draves, R., “Privacy Extensions for Stateless Address Autoconfiguration in IPv6”, RFC 3041, January 2001
10. Montenegro, G., Castelluccia, C., “SUCV Identifiers and addresses”, Internet draft, Work in progress, November 2001
11. Dommety, G., "Fast Handovers for Mobile IPv6", Internet draft, Work in progress, 2001.
12. Wasserman, M., “Recommendations for IPv6 in 3GPP Standards”, Internet Draft, Work in progress, April 2002

Appendix A: The Birthday Problem

The classical formulation of the birthday problem is as follows: we want to calculate the probability that in a group of k people, al least two of them have the same birthday.

We model the birthday as a integer random variable, with uniform distribution between 1 and n (in this particular case n is the number of possible birthdays i.e. n=365)

Then, the number N of ways that we can choose k values out of n with no duplicates would be:

$$N = n \cdot (n-1) \cdot \ldots \cdot (n-k+1)$$

On the other hand, the number of possibilities for choosing k elements out of n, without the restriction of not having any duplicates is n^k

Then, the probability of not having a collision when we select k elements out of n is:

$$P_{NO}(n,k) = \frac{n!}{(n-k)! \cdot n^k}$$

So, the probability of having at least one collision when we select k elements out of n is:

$$P(n,k) = 1 - \frac{n!}{(n-k)! \cdot n^k}$$